

# Complete Characterization of Stability of Cluster Synchronization in Complex Dynamical Networks

Francesco Sorrentino<sup>1</sup>, Louis M. Pecora<sup>2</sup>, Aaron M. Hagerstrom<sup>3,4</sup>, Thomas E. Murphy<sup>4,5</sup>, Rajarshi Roy<sup>3,4,6</sup>

1. *Department of Mechanical Engineering, University of New Mexico, Albuquerque, NM 87131*

2. *U. S. Naval Research Laboratory, Washington, DC 20375*

3. *Department of Physics, University of Maryland, College Park, MD 20742*

4. *Institute for Research in Electronics and Applied Physics,  
University of Maryland, College Park, MD 20742*

5. *Department of Electrical and Computer Engineering,  
University of Maryland, College Park, MD 20742 and*

6. *Institute for Physical Science and Technology,  
University of Maryland, College Park, MD 20742*

(Dated: December 4, 2015)

Synchronization is an important and prevalent phenomenon in natural and engineered systems. In many dynamical networks, the coupling is balanced or adjusted in order to admit global synchronization, a condition called Laplacian coupling. Many networks exhibit incomplete synchronization, where two or more clusters of synchronization persist, and computational group theory has recently proved to be valuable in discovering these cluster states based upon the topology of the network. In the important case of Laplacian coupling, additional synchronization patterns can exist that would not be predicted from the group theory analysis alone. The understanding of how and when clusters form, merge, and persist is essential for understanding collective dynamics, synchronization, and failure mechanisms of complex networks such as electric power grids, distributed control networks, and autonomous swarming vehicles. We describe here a method to find and analyze all of the possible cluster synchronization patterns in a Laplacian-coupled network, by applying methods of computational group theory to dynamically-equivalent networks. We present a general technique to evaluate the stability of each of the dynamically valid cluster synchronization patterns. Our results are validated in an electro-optic experiment on a 5 node network that confirms the synchronization patterns predicted by the theory.

## I. INTRODUCTION

Synchronization of oscillators in large networks has been an interesting problem for many years. It's a phenomenon that shows up in many natural and man-made systems [1–3]. A particular type of synchronization is global synchronization where the oscillators all follow the same trajectory in state space. A network where this is desirable would be generators in a power grid, as well as some control networks and swarming autonomous vehicles. Although global synchronization has a well-developed theory [4–6] a more recently studied, more complex phenomenon we will call cluster synchronization, which we will abbreviate as CS, has attracted considerable attention [7–16]. In CS the network evolves into subsets of oscillators in which members of the same cluster are synchronized to the same trajectory, but members of different clusters are not. Such synchronized clusters may show up in swarms of animals, where the network is the simple visual link to one's neighbors, or swarms of unmanned autonomous vehicles which are connected by a local communication network. Clusters may also show up in power grids where they would be a sign of a problem, that is, loss of global synchronization.

Given the increase in man-made networks and the growing use of network theory to describe natural systems (e.g., food webs, neuronal and genetic networks) it is important to develop a basic approach to determining what cluster structures are possible in a given network. In this paper we show what methods can be used to do this using the concept of oscillators coupled through connection to other (not necessarily all) oscillators in the network. More importantly, we extend these methods to the currently unsolved problem of finding clusters in networks of oscillators which have a self-coupling to balance incoming signals from other oscillators, often called Laplacian coupling (more on this below). These allow synchronization clusters that elude other methods to find cluster patterns. We show how to use and extend symmetry methods to find all possible clusters in such networks of Laplacian-coupled oscillators. We first show how network symmetries can lead to cluster synchronization. We then show how we can go beyond this to analyze CS resulting from Laplacian Coupling.

In Sec. II we review the concepts of symmetries and clusters in networks of coupled oscillators. In Sec. III we discuss methods to uncover all of the possible cluster synchronization patterns in a given network. Our main results are contained in Sec. IV, where we present a stability analysis that applies to any cluster synchronization pattern. We also present an electro-optic experiment that confirms the patterns of synchronization predicted by the theory. Finally, the conclusions are presented in Sec. V.

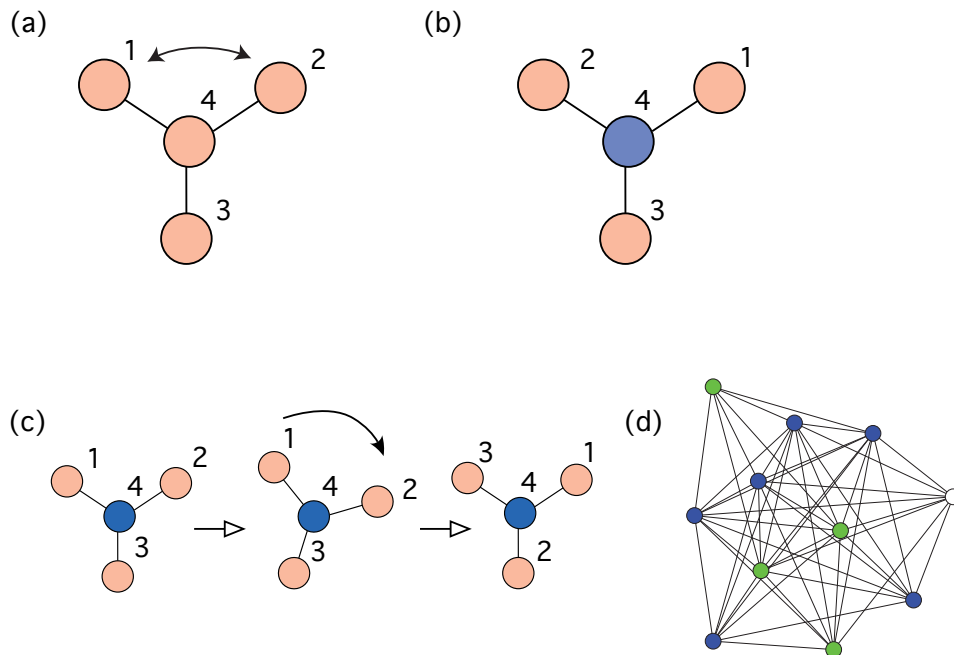


FIG. 1. (color online) (a) A network of 4 identical oscillators coupled through 3 identical links, (b) the same network after a reflection operation, (c) the same network after a rotation operation, (d) an 11-node network showing 3 clusters (blue, green, white).

## II. SYMMETRIES AND CLUSTERS IN NETWORKS

Fig. 1 (a) shows a 4-oscillator or 4-node network, where the oscillators are identical as are the couplings between them, which are bidirectional, meaning that signals flow in both directions to the oscillators by the same amount and influence on the connected oscillators. This network has a total of 6 symmetries. We show two. In Fig 1 (b) we show the result of a reflection (shown in (a)) interchanging nodes 1 and 2. The structure of the network remains the same. In Fig 1 (c) we show a rotation of the network by  $120^\circ$  which also leaves the network indistinguishable from the original in (a). These symmetries are manifest in the symmetries of a matrix that describes the network, the adjacency matrix. This matrix contains a 1 in each position of each row for each node that is connected to the node corresponding to the row number. For the network in Fig. 1 (a) the adjacency matrix  $A$  is,

$$A = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \quad (1)$$

The adjacency matrix plays a crucial role in modeling the dynamics of many networks of identical nodes, as it provides the coupling between the oscillators at each node. We will write the dynamics of the networks as follows,

$$\dot{\mathbf{x}}_i = \mathbf{F}(\mathbf{x}_i) + \sigma \sum_j A_{ij} \mathbf{H}(\mathbf{x}_j), \quad (2)$$

$i = 1, \dots, N$ , where  $\mathbf{x}_i$  is the vector of dynamical variables for the  $i^{th}$  oscillator,  $\dot{\mathbf{x}}_i$  is the time derivative of the  $i$ th node's variables,  $N$  is the number of oscillators (nodes),  $\mathbf{F}$  is the vector field of each node (governing each oscillator's isolated dynamics), and  $\mathbf{H}$  is a coupling function for each link in the network of each oscillator to another. The entries  $A_{ij}$  represent the strength of the coupling from node  $j$  to node  $i$ . Several papers [8, 11, 12, 15, 17–20] have used Eq. (2) to model the dynamics of a network.

A network symmetry applied to the adjacency matrix leaves it unchanged. We recall that symmetries of an object (a network in this case) form a mathematical group  $\mathcal{G}$ . Any element in this group, say  $g$  is represented in the space of network nodes as a permutation matrix, say  $R_g$ . The invariance of  $A$  under the action of the symmetry immediately implies  $R_g A = A R_g$ ,  $A$  commutes with all group actions and the equations of motion for nodes that are mapped into each other are the same. For example, in the network in Fig. 1 (a), nodes 1,2, and 3 have the same equations of motion, so if they are started from the same initial conditions, they will remain synchronized indefinitely. Node 4 cannot be permuted into any of the other nodes and it will not synchronize with the others, hence we color it differently in the Figure parts (b) and (c). This is the intimate relationship between symmetry and dynamics in networks. We see that it immediately separates the network into two clusters 1,2,3 and 4.

Many of the recent studies of cluster synchronization are on particular networks that are known or engineered to exhibit cluster synchrony. But as we showed above group theory provides a general approach to finding clusters in arbitrary networks. Steps toward more general approaches using group theory to analyze cluster synchronization began with the work of Golubitsky, Stewart, and Schaeffer [21, 22], where network symmetries are known and can be shown to support CS. Recently, computational methods have been used to study simple symmetric networks and an approach has been developed that relates the symmetries with the emergence of the CS states [23]. Finally, we showed that such approaches can be applied to more complex networks with hundreds of oscillators using computational group theory [24].

An alternative description of the network dynamics is the following in the case of Laplacian coupling,

$$\dot{\mathbf{x}}_i = \mathbf{F}(\mathbf{x}_i) + \sigma \sum_j A_{ij} [\mathbf{H}(\mathbf{x}_j) - \mathbf{H}(\mathbf{x}_i)], \quad (3)$$

where the coupling from oscillator  $j$  to oscillator  $i$  is given by the the difference between the output functions  $\mathbf{H}(\mathbf{x}_j)$  and  $\mathbf{H}(\mathbf{x}_i)$ . Several papers [2, 5, 7, 10, 20, 23, 25] have used Eq. (3) to model the dynamics of a network. Equation (3) can be rewritten as follows,

$$\dot{\mathbf{x}}_i = \mathbf{F}(\mathbf{x}_i) + \sigma \sum_j L_{ij} \mathbf{H}(\mathbf{x}_j), \quad (4)$$

which has the same structure as Eq. (2) but now the adjacency matrix has been replaced by the Laplacian matrix,  $L = \{L_{ij}\}$ ,  $L_{ij} = A_{ij} - \delta_{ij} \sum_j A_{ij}$ , where  $\delta_{ij}$  is the Kronecker delta. By construction then we have that the sums of the rows of the matrix  $L$  are equal to zero, i.e. the inputs to the  $i$ th node are balanced by the diagonal self-coupling.

In Ref. [26] Golubitsky, Stewart, and Török have shown that for networks that have balanced coupling (all nodes receive the same cumulative input weights, accounting for adjacent nodes and self-coupling, an example of which is Eq. (4)), CS can emerge in many patterns that are not directly the result of symmetries. An example of this is global synchronization, which is not a result of symmetries in the network.

In general, the patterns of cluster synchrony that can be observed in a network are not unique [27, 28], hence an important problem is that of determining the parameter ranges for stability and multi-stability for the observed patterns. While it is known that stability of the global synchronization state for an arbitrary network can be characterized by using the master stability function formalism [29], a corresponding analysis that applies to the CS patterns that may emerge in a network is not available. In this paper we address this problem by providing necessary and sufficient conditions for stability of each individual CS pattern under very general assumptions. Our analysis applies to systems for which the functions describing the individual dynamics (possibly chaotic) and the interactions between the systems are arbitrary and to both the cases that the network is described by an adjacency matrix (Eq. (2)) or by a Laplacian matrix (Eq. (4)), both of which are used to model network interactions.

In what follows we show how to find all of the CS patterns that may emerge in a given network topology, described either by an adjacency matrix or by a Laplacian matrix and how to evaluate the stability of each allowed pattern. We demonstrate all the above phenomena in an electro-optic experiment on a 5 node network that displays all of the possible CS patterns predicted by the theory.

### III. ANALYZING CLUSTER SYNCHRONIZATION PATTERNS

Here we attempt at addressing the following problem: given a network structure (either in terms of an adjacency matrix in Eq. (2) or of a Laplacian matrix in Eq. (4)) can we find all of the cluster synchronization (CS) patterns that are allowed? For simplicity, we will proceed under the assumption that the network dynamics is described by Eq. (4) but all of our results include the simpler case that the dynamics is described by Eq. (2). Indeed, as we will see, the case of the Laplacian matrix is in general more complex to deal with than that of the adjacency matrix. So we consider the most difficult case.

First of all, we note that a symmetry of the adjacency matrix is also a symmetry of the corresponding Laplacian matrix and viceversa [24]. Suppose  $\mathcal{G}$  is a group of permutations of the nodes of the network which leaves the coupling matrix  $L$  invariant. Then for each  $g \in \mathcal{G}$  we have a permutation matrix  $R_g$  that operates on the set of all node vectors  $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_N)^T$ . Since  $R_g L = L R_g$ , this means  $d(R_g \mathbf{x}_i)/dt = \mathbf{F}(R_g \mathbf{x}_i) + \sigma \sum_j L_{ij} \mathbf{H}(R_g \mathbf{x}_j)$ , i.e., the symmetry operation leaves the equations of motion unchanged. Hence, the subset of nodes permuted among each other by the group will remain synchronized if started in a synchronized state. We will refer to these subsets of nodes as clusters. The synchronized states for each cluster are flow invariant.

An approach to the construction of all allowed CS patterns in a network has been proposed in [30]. While the method presented in [30] is general, as it applies to any network topology, it is computationally expensive. Here we argue that for the case of symmetric networks, a faster approach can often be followed that takes advantage of computational group theory, which is quite efficient. At each step we show the results of this method applied to a particular case.

We first decompose the symmetry group  $\mathcal{G}$  into subgroups  $\mathcal{H}_i, i = 1, \dots, \nu$ , each of which acts only on some subset of clusters (often only one), but not on any of the others [31, 32]. For this reason, we will refer to these subgroups as cluster groups and the original group is a direct product of all cluster groups  $\mathcal{G} = \mathcal{H}_1 \times \dots \times \mathcal{H}_\nu$ . We further decompose each cluster group  $\mathcal{H}_i$  into all of its possible subgroups, which will give us a natural set of symmetry-breaking paths. These subgroups provide the full range of possible symmetry clusters from the original full symmetry clusters to subclusters to the trivial case where each node is in its own cluster, i.e. no symmetries. While for the case of the adjacency matrix, this allows one to find all of the possible CS patterns, in the case of the Laplacian matrix, these patterns are certainly valid but there may be other valid patterns that are not predicted by the computational group theory analysis. Extra steps are thus required in order to find these additional patterns.

We examine a particular case, a 5-node network (Fig. 2), with adjacency matrix

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix} \quad (5)$$

and Laplacian matrix

$$L = \begin{pmatrix} -3 & 1 & 0 & 1 & 1 \\ 1 & -3 & 1 & 0 & 1 \\ 0 & 1 & -3 & 1 & 1 \\ 1 & 0 & 1 & -3 & 1 \\ 1 & 1 & 1 & 1 & -4 \end{pmatrix} \quad (6)$$

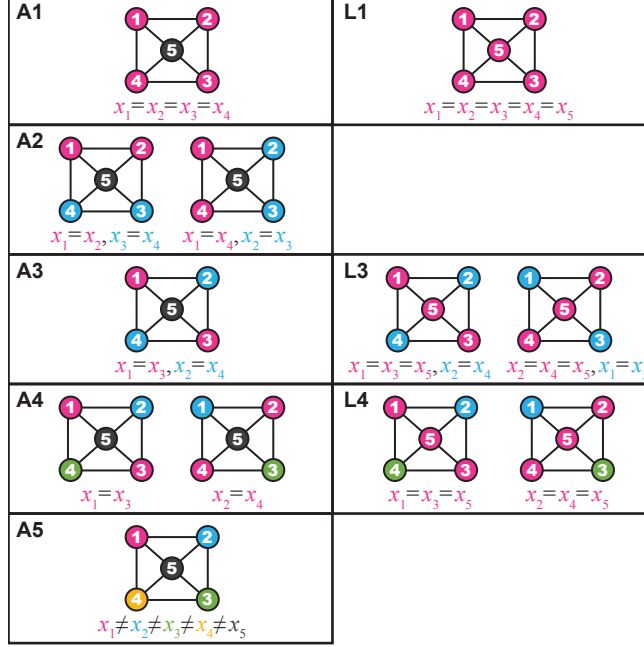


FIG. 2. (color online) On the left: **all possible** patterns displayed when the network connectivity is given by the adjacency matrix (5). On the right: additional patterns displayed when the network connectivity is given by the Laplacian matrix (6).

Figure 2 shows all the allowed patterns, where nodes belonging to the same cluster are colored the same.

Cluster patterns A1 to A5 in Fig. 2 can be found by performing an analysis of the group  $\mathcal{G}$  and all of its subgroups. These are all the patterns that may emerge for the adjacency matrix (5) and the only patterns that may emerge by symmetry for the Laplacian matrix (6). The orbits of the original symmetry group are  $\{1, 2, 3, 4\}$  and  $\{5\}$  by itself and are associated with two cluster groups:  $\mathcal{H}_1$  which permutes the first and  $\mathcal{H}_2$  which is only the identity for the 2nd single node cluster. Other patterns (A2-A5) are possible based on subgroups of the original group. These are the result of symmetry breaking bifurcations.

Now we create new potential clusters by first, choosing a set of cluster groups ( $\mathcal{H}_i$ ) and/or their subgroups (one for each cluster group). Together these determine a subgroup  $\mathcal{G}'$  of the original group  $\mathcal{G}$ . Second, we combine or merge some of these clusters as candidates for new synchronized clusters (it's our choice which to try). Because we are merging clusters or subclusters from different original symmetry clusters, the resulting CS patterns will not be the result of symmetries but may be dynamically valid when the coupling is Laplacian. Third, we can set those dynamical variables  $\mathbf{x}_i$  equal to others in the merged cluster and see if their equations of motion are the same (guaranteeing flow invariance). However, examining equations of motion by eye for large networks can become prohibitive.

There is a more direct way of doing step three above using the power and efficiency of group theory and computational discrete algebra tools [33, 34]. Examining the coupling term in Eq. (4) when clusters are synchronized the diagonal feedback term ( $\mathbf{H}(\mathbf{x}_i)$ ) of a node will cancel the coupling terms of nodes from the same merged cluster. Hence, in the synchronized state the dynamics behaves as though nodes from the same merged cluster are not connected. We therefore form a dynamically equivalent coupling matrix which is the original Laplacian matrix with off-diagonal components between nodes from merged clusters set to 0 and the diagonal values then set to the negative of the new row sums. We then perform the cluster group decompositions and subgroup constructions on the new, dynamically equivalent Laplacian (note that if the original Laplacian matrix is symmetric, so are also the dynamically equivalent matrices obtained through this construction). If some set of subgroups of this new coupling matrix has symmetries yielding clusters which are our merged clusters, then their dynamics is flow invariant in the synchronized

state and the cluster merging is possible. In this case we call the new clusters Laplacian clusters. All the above can be automated in software. Hence, in Laplacian networks, even when a CS pattern is not the direct result of a symmetry of the original coupling matrix, it is the result of a symmetry of a dynamically equivalent coupling matrix. This is particularly relevant in terms of computational complexity. In fact, while the problem of finding all of the symmetry operations of a given matrix has not been proven to be polynomial, efficient discrete algebra routines have been developed that make these computations possible even for very large networks (e.g., the Internet at the AS level, for which  $N = 22332$ , see Ref. [32]). This is in contrast with the method proposed in [30], not based on evaluation of the symmetries, which has been shown to become inefficient even for networks of moderate size, e.g.,  $N = 15$ .

In Fig. 2 the only possibility of merging is for node 5 to be absorbed into one of the remaining clusters in patterns  $A_1, A_2, \dots, A_5$ , which is shown on the right hand side. Analyzing our dynamically equivalent matrix as above, we find three additional allowed CS patterns,  $L_1, L_3$ , and  $L_4$ . These are not predicted by the original symmetry analysis but can emerge when the coupling matrix is Laplacian. Note that the potential  $L_2$  patterns from merging of the center with one of the other clusters in A2 will not lead to a new synchronized cluster (the center has two inputs from the unsynchronized cluster while the corners have only one). The group analysis of the dynamically equivalent network agrees with this.

To conclude, when the network is described by an adjacency matrix, a full characterization of all the dynamically valid patterns can be obtained by taking advantage of available computational discrete algebra tools [33, 34]. These output the cluster groups,  $\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_\nu$  and a decomposition of each cluster group  $\mathcal{H}_i$  into all of its possible subgroups, which provides a natural set of symmetry-breaking paths. As discussed above, available computational group theory routines perform these tasks very efficiently, when compared to other possible methods (e.g., [30]). For this case our approach is always better than the state of the art. When the coupling is in Laplacian form, all of the (symmetry-related) CS patterns of the associated adjacency matrix are maintained. Moreover, it is possible for some of the cluster groups  $\mathcal{H}_1, \mathcal{H}_2, \dots, \mathcal{H}_\nu$  and some of their subgroups to merge to form new dynamically valid clusters. In order to find all of the dynamically valid mergings, it can be helpful to use the clusters and subclusters provided by the symmetry analysis as the building blocks of our algorithm. If the number of these different clusters and subclusters is equal to  $\mu$ , the number of tests that need to be performed (for pairs, triplets, and so on) is upper bounded by  $B_\mu$ , the  $\mu$ -th Bell number, hence it grows combinatorially with  $\mu$ . For all the networks that do not display many symmetries, i.e., for which  $\mu \ll N$ , our approach based on computational group theory will be much faster than the one presented in [30]. Hence, in general it is a good idea to preliminarily run an analysis of the symmetries of the network to assess whether  $\mu < N$  and choose the most convenient approach based on this outcome. It should be noted that there is no need for testing for mergings between subgroups of the same group, which reduces the complexity of the calculations. In general, finding all of the dynamically valid patterns for Laplacian networks may be substantially harder than for networks described by a symmetric adjacency matrix. Similar limitations were observed in [30].

#### IV. STABILITY ANALYSIS AND EXPERIMENTAL VALIDATION

Stability of cluster synchronization has been investigated for phase oscillators [23] and Stuart-Landau oscillators [35], and for lattices of coupled systems [27, 28]. However, a general approach to analyze stability and multistability of CS patterns in arbitrary networks has not been developed. In [24] we studied a particular CS pattern corresponding to a minimum number of clusters (i.e., maximal symmetry) for the case that the connectivity of the network is in the form of an adjacency matrix.

We now develop variational equations for the merged-cluster system so we can calculate the stability of each one of the allowed CS patterns. While a number of papers [23, 26–28, 35–37] have dealt with cluster synchronization in networks, only Refs. [23, 35] have considered the problem of stability for particular dynamics of the individual systems. Refs. [27, 28, 37] have emphasized that for arbitrary systems this is a difficult problem. In order to analyze stability, we start with the subgroup  $\mathcal{G}'$  of the original group which generated the clusters we want to merge. It is formed by a direct product of the subgroups we choose to use in our merged system. Using  $\mathcal{C}_m$  to represent each cluster of nodes,  $m = 1, \dots, K$ , where  $K$ =number of clusters in  $\mathcal{G}'$ , we have the variational equation of Eq. (2),

$$\delta \dot{\mathbf{x}}(t) = \left[ \sum_{m=1}^M E^{(m)} \otimes D\mathbf{F}(\mathbf{s}_m(t)) + \sigma \sum_{m=1}^M (L E^{(m)}) \otimes D\mathbf{H}(\mathbf{s}_m(t)) \right] \delta \mathbf{x}(t), \quad (7)$$

where the  $Nn$ -dimensional vector  $\delta \mathbf{x}(t) = [\delta \mathbf{x}_1(t)^T, \delta \mathbf{x}_2(t)^T, \dots, \delta \mathbf{x}_N(t)^T]^T$ ,  $L$ = the Laplacian-coupling matrix, and

Pattern	Quotient Network Dynamics (right hand side is mod $2\pi$ )	Transverse Perturbations
A1: $x_1 = x_2 = x_3 = x_4$	$x_1^{t+1} = [(\beta - \sigma)\mathcal{I}(x_1^t) + \sigma\mathcal{I}(x_5^t) + \delta]$ $x_5^{t+1} = [(\beta - 4\sigma)\mathcal{I}(x_5^t) + 4\sigma\mathcal{I}(x_1^t) + \delta]$	$\eta^{t+1} = [\beta + \sigma\lambda] D\mathcal{I}(x_1^t)\eta^t, \lambda = -3, -5$
A2: $x_1 = x_2 \quad \& \quad x_3 = x_4$	$x_1^{t+1} = [(\beta - 2\sigma)\mathcal{I}(x_1^t) + \sigma\mathcal{I}(x_4^t) + \sigma\mathcal{I}(x_5^t) + \delta]$ $x_2^{t+1} = [(\beta - 3\sigma)\mathcal{I}(x_2^t) + 3\sigma\mathcal{I}(x_1^t) + \delta]$ $x_4^{t+1} = [(\beta - 3\sigma)\mathcal{I}(x_4^t) + 3\sigma\mathcal{I}(x_1^t) + \delta]$	$\eta_1^{t+1} = [\beta - 4\sigma] D\mathcal{I}(x_1^t)\eta_1^t - \sigma D\mathcal{I}(x_3^t)\eta_3^t$ $\eta_3^{t+1} = -\sigma D\mathcal{I}(x_1^t)\eta_1^t + [\beta - 4\sigma] \sigma D\mathcal{I}(x_3^t)\eta_3^t$
A3: $x_1 = x_3 \quad \& \quad x_2 = x_4$	$x_1^{t+1} = [(\beta - 2\sigma)\mathcal{I}(x_1^t) + \sigma\mathcal{I}(x_2^t) + \sigma\mathcal{I}(x_5^t) + \delta]$ $x_2^{t+1} = [(\beta - 2\sigma)\mathcal{I}(x_2^t) + \sigma\mathcal{I}(x_1^t) + \sigma\mathcal{I}(x_5^t) + \delta]$ $x_5^{t+1} = [(\beta - 4\sigma)\mathcal{I}(x_5^t) + 2\sigma\mathcal{I}(x_1^t) + 2\sigma\mathcal{I}(x_2^t) + \delta]$	$\eta^{t+1} = [\beta + \lambda\sigma] D\mathcal{I}(x_1^t)\eta^t, \lambda = -3$
A4: $x_1 = x_3$	$x_1^{t+1} = [(\beta - 3\sigma)\mathcal{I}(x_1^t) + \sigma\mathcal{I}(x_2^t) + \sigma\mathcal{I}(x_4^t) + \sigma\mathcal{I}(x_5^t) + \delta]$ $x_2^{t+1} = [(\beta - 3\sigma)\mathcal{I}(x_2^t) + 2\sigma\mathcal{I}(x_1^t) + \sigma\mathcal{I}(x_5^t) + \delta]$ $x_4^{t+1} = [(\beta - 3\sigma)\mathcal{I}(x_4^t) + 2\sigma\mathcal{I}(x_1^t) + \sigma\mathcal{I}(x_5^t) + \delta]$ $x_5^{t+1} = [(\beta - 4\sigma)\mathcal{I}(x_5^t) + 2\sigma\mathcal{I}(x_1^t) + \sigma\mathcal{I}(x_2^t) + \sigma\mathcal{I}(x_4^t) + \delta]$	$\eta^{t+1} = [\beta + \lambda\sigma] D\mathcal{I}(x_1^t)\eta^t, \lambda = -3$
L1: $x_1 = x_2 = \dots = x_5$	$x_1^{t+1} = [\beta\mathcal{I}(x_1^t) + \delta]$	$\eta^{t+1} = [\beta + \sigma\lambda] D\mathcal{I}(x_1^t)\eta^t, \lambda = -3, -5$
L3: $x_1 = x_3 = x_5 \& x_2 = x_4$	$x_1^{t+1} = [(\beta - 2\sigma)\mathcal{I}(x_1^t) + 2\sigma\mathcal{I}(x_2^t) + \delta]$ $x_2^{t+1} = [(\beta - 3\sigma)\mathcal{I}(x_2^t) + 3\sigma\mathcal{I}(x_1^t) + \delta]$	$\eta_1^{t+1} = [\beta + \sigma\lambda] D\mathcal{I}(x_1^t)\eta_1^t, \lambda = -3, -5$ $\eta_2^{t+1} = [\beta + \sigma\lambda] D\mathcal{I}(x_2^t)\eta_2^t, \lambda = -3$
L4: $x_1 = x_3 = x_5$	$x_1^{t+1} = [(\beta - 2\sigma)\mathcal{I}(x_1^t) + \sigma\mathcal{I}(x_2^t) + \sigma\mathcal{I}(x_4^t) + \delta]$ $x_2^{t+1} = [(\beta - 3\sigma)\mathcal{I}(x_2^t) + 3\sigma\mathcal{I}(x_1^t) + \delta]$ $x_4^{t+1} = [(\beta - 3\sigma)\mathcal{I}(x_4^t) + 3\sigma\mathcal{I}(x_1^t) + \delta]$	$\eta^{t+1} = [\beta + \sigma\lambda] D\mathcal{I}(x_1^t)\eta^t, \lambda = -3, -5$

TABLE I. Equations used to evaluate stability of all the allowed CS patterns for the system described by Eq. (10) and coupling matrix (6).

$E^{(m)}$  is an  $N$ -dimensional diagonal indicator matrix for each cluster such that  $E_{ii}^{(m)}$  is equal to 1 if  $i \in \mathcal{C}_m$  and is equal to 0, otherwise,  $i = 1, \dots, N$ . Note that we must use the original Laplacian matrix and not the dynamically equivalent one, which is used only for detecting synchronization flow invariance.

As we showed in [24] we can first block diagonalize the coupling matrix  $L$  using the irreducible representations (IRR) of  $\mathcal{G}'$ , which yields the transformed coupling matrix  $L'$  for A3 of Fig. 2,

$$L' = \begin{pmatrix} -4 & -\sqrt{2} & \sqrt{2} & 0 & 0 \\ -\sqrt{2} & -3 & -2 & 0 & 0 \\ \sqrt{2} & -2 & -3 & 0 & 0 \\ 0 & 0 & 0 & -3 & 0 \\ 0 & 0 & 0 & 0 & -3 \end{pmatrix} \quad (8)$$

The upper-left block represents the variations on the synchronization manifold. It is  $3 \times 3$  because there are three different trajectories in A3 (three clusters). The lower-right block represents the variations in the transverse manifold. These are the perturbations that take the system out of synchrony so it is these whose stability we want to calculate. Suppose we merge the center node (5) with nodes 1 and 3 to form the first merged state shown in L3. Geometrically, what must happen is that the dimension of the synchronization manifold must decrease by one (from 3 to 2) and the transverse manifold must increase by 1 (from 2 to 3).

To obtain the new coordinates on the synchronization manifold, we note that basis vectors of a cluster on the synchronization manifold have a 1 in the position of each node that belongs to the cluster. For example, in A3 the cluster (1,3) will have a (unit) vector of the form (1,0,1,0,0) and (5) will have the (unit) vector of the form (0,0,0,0,1). The merged cluster (1,3,5) will have the synchronization manifold vector which is their sum, (1,0,1,0,1). The transformation of this new synchronization vector to the IRR coordinates of  $L'$  provides the new synchronization direction and its orthogonal complement provides the new transverse direction. We use these two new vectors to transform the  $2 \times 2$  sub-block associated with the (1,3) and (5) clusters in the  $3 \times 3$  synchronization block to reduce the  $3 \times 3$  synchronization manifold and increase the transverse manifold. This results in the final variational equation for the L3 case,

$$\dot{\boldsymbol{\eta}} = \sum_{m=1}^M \left[ J^{(m)} \otimes D\mathbf{F}(\mathbf{s}_m) + \sigma L'' J^{(m)} \otimes D\mathbf{H}(\mathbf{s}_m) \right] \boldsymbol{\eta},$$



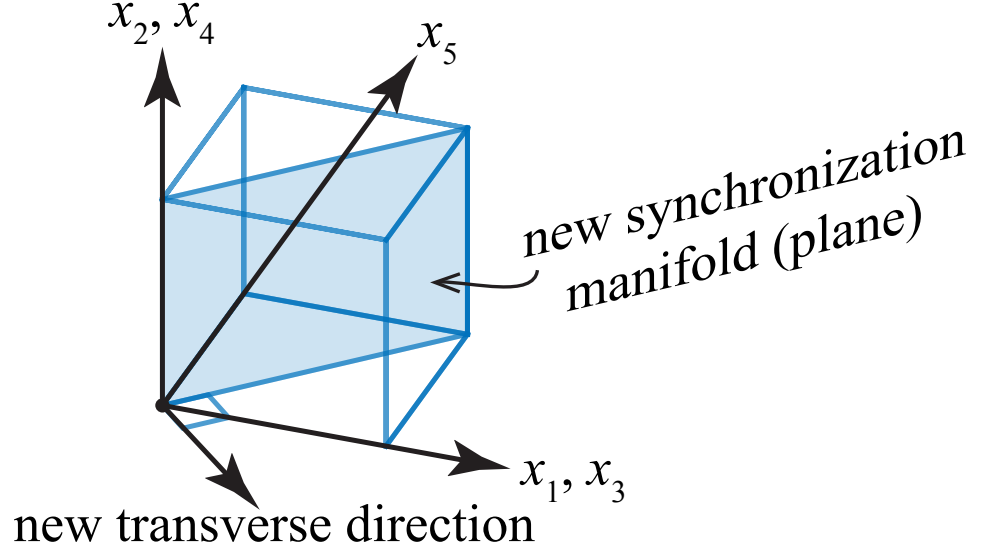


FIG. 3. Reduction of the dimension of the three-dimensional synchronization manifold. This shows schematically how the merging of clusters (1,3) and (5) produces a new synchronization direction in the (1,3) and (5) plane of the synchronization manifold along with a new transverse direction orthogonal to the new synchronization direction.

where we have linearized about the new synchronized merged cluster states  $\{\mathbf{s}_1, \dots, \mathbf{s}_M\}$ ,  $\boldsymbol{\eta}$  is the vector of variations of all nodes transformed to the merged coordinates as above,  $D\mathbf{F}$  and  $D\mathbf{H}$  are the Jacobians of the nodes' vector field and coupling function, respectively,  $J^{(m)}$  are the transformed  $E^{(m)}$  and,

$$L'' = \begin{pmatrix} -3 & \sqrt{6} & 0 & 0 & 0 \\ \sqrt{6} & -2 & 0 & 0 & 0 \\ 0 & 0 & -5 & 0 & 0 \\ 0 & 0 & 0 & -3 & 0 \\ 0 & 0 & 0 & 0 & -3 \end{pmatrix}, \quad (9)$$

In Eq.(9) the new synchronization block (in the upper-left-hand corner) represents the new clusters (1,3,5) and (2,4) and the new transverse direction is associated with the new diagonal value  $-5$ . Obviously, this can be generalized to more complex cluster mergings. An example of cluster merging is shown geometrically in Fig. 3. We remark that this method can be used to analyze stability of any dynamically valid CS pattern, for which knowledge of the block-diagonalized matrix  $L'$  is available and it applies to both cases that the connectivity is given by an adjacency matrix or by a Laplacian matrix. The generalization of the above procedure simply uses the synchronization vectors for all the new clusters as the rows of a matrix for which all other components are zero. An application of a singular value decomposition then gives a basis for the synchronization block (the original synchronization vectors) and a basis for the orthogonal complement which represents all the transverse directions. In some cases the latter can be simplified by a block diagonalization when all members of the block are in the same merged cluster. This makes it possible to automate this block-diagonalization to evaluate stability of all the CS patterns that can emerge in a given network topology.

We show symmetry-breaking and the existence of Laplacian clusters using the experimental system described in detail in [24, 38]. This system employs a spatial light modulator (SLM) and a camera in a feedback configuration. The camera has a focal plane array (FPA) of  $320 \times 256$  pixels, and an area of  $8 \times 6.4 \text{ mm}^2$ . The SLM has a resolution of  $512 \times 512$  pixels, and an active area of  $7.68 \times 7.68 \text{ mm}^2$ . A light emitting diode with a wavelength of  $1550 \text{ nm}$  is used to illuminate the modulator. The light passes through a polarizing beam-splitter, and a quarter-wave plate (QWP), so that circularly polarized light is incident on the SLM. The SLM imparts a programmable spatially-varying phase shift  $x$  between the two polarization components of the reflected light. The reflected light passes through the QWP and polarizer, and is imaged onto a  $256 \times 256$  pixel square region of the camera's FPA. The relationship between the phase shift  $x$  applied by the SLM and the normalized intensity  $\mathcal{I}$  recorded by the camera is  $\mathcal{I}(x) = (1 - \cos x)/2$ .

Each oscillator corresponds to a square patch of  $16 \times 16$  pixels on the SLM, which is imaged onto an  $8 \times 8$  pixel



region of the camera's FPA. Using a computer, the phase shift of the  $i$ th region,  $x_i$ , is iteratively updated based on the intensity measured by the camera according to:

$$x_i^{t+1} = \left[ \beta \mathcal{I}(x_i^t) + \sigma \sum_j L_{ij} \mathcal{I}(x_j^t) + \delta \right] \bmod 2\pi \quad (10)$$

where  $\sigma$  is the coupling strength, which we will vary. Eq. (10) is a map version of Eq. (2) and can show fixed point, periodic or chaotic dynamics depending on the values of the parameters. Here,  $\delta = 0.525$ ,  $\beta = 1.45\pi$ , and  $\sigma$  was decreased from  $\pi$  to 0.

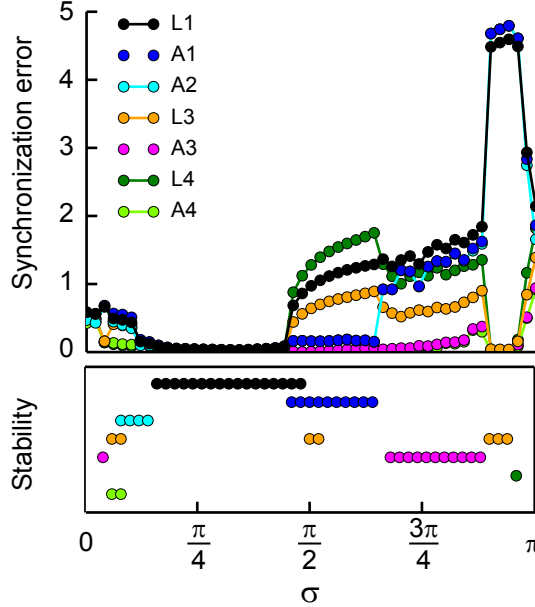


FIG. 4. The figure shows the experimental synchronization error for each synchronization pattern as a function of the parameter  $\sigma$ . Underneath we plot the results of our stability analysis applied to each one of the cluster synchronization patterns, where a colored dot labels the values of sigma for which the corresponding pattern is stable.

Fig. 4 shows the experimental synchronization error for each synchronization pattern shown in Fig. 2 as a function of  $\sigma$ . The phases  $x_i$  were not reinitialized when  $\sigma$  was updated. The synchronization error for each pattern is computed as  $\langle |x_i(t) - x_i^{cl}(t)| \rangle$ , where the symbol  $\langle \bullet \rangle$  indicates an average both in time and over the nodes  $i$  in a cluster and the clusters in a pattern.  $x_i^{cl}$  is the average state for the nodes in the cluster to which node  $i$  belongs. In the lower panel of Fig. 4, we plot the results of our stability analysis for each one of the CS patterns, where a colored dot labels the values of sigma for which the corresponding pattern is stable. In particular, a CS pattern was indicated to be stable when (i) all the numerically computed maximum Lyapunov exponents corresponding to the transverse blocks were found to be negative and (ii) the synchronous pattern was asymptotically valid, that is the CS pattern was observed after integrating its equations for a long time. The equations that were used to run these stability calculations are shown in Table I.

In general, when two or more clusters merge into one, there are two independent effects on stability, as can be seen from the structure of the block-diagonalized matrix  $L''$ : the first one is that the dimension of the synchronization block decreases, which determines the motion in the synchronization manifold; the second one is that new transverse blocks appear, with the other (pre-existing) transverse blocks remaining the same. As a consequence, since each one of these transverse blocks needs to be individually tested for stability, we expect that as the number of transversal blocks increases, the range of stability decreases. This is confirmed by our experimental results plotted in Fig. 4, showing that the  $\sigma$ -range of stability becomes smaller for CS patterns that are characterized by higher symmetry. Exceptions to this rule are possible as the motion in the synchronization manifold (on which the transverse Lyapunov exponents depend) may also affect stability in ways that cannot be predicted by the analysis of the transverse blocks only.

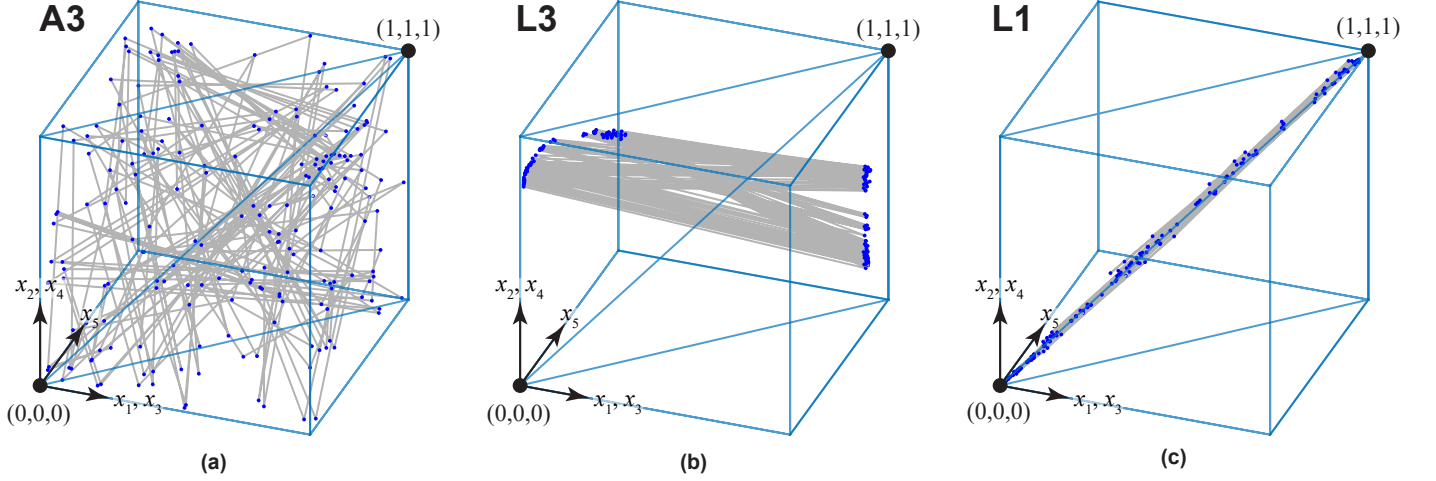


FIG. 5. Experimental phase space plots with lines connecting successive iterates for a) three clusters (A3), b) two clusters (L3) and c) one cluster (L1).

Fig. 5 shows the dynamics on the synchronization manifold for the symmetry pattern A3 and the two merged patterns, L3 and L1. In each transition  $A3 \rightarrow L3 \rightarrow L1$  the dimension of the synchronization manifold decreases by 1 (from 3 to 2 to 1) and the transverse manifold increases by 1 (from 2 to 3 to 4). Fig. 6 shows three snapshots of the experimental dynamics for each one of the patterns A3, L3, and L1. Videos of the synchronization manifolds and patterns of Fig. 6 are available in the supplement.

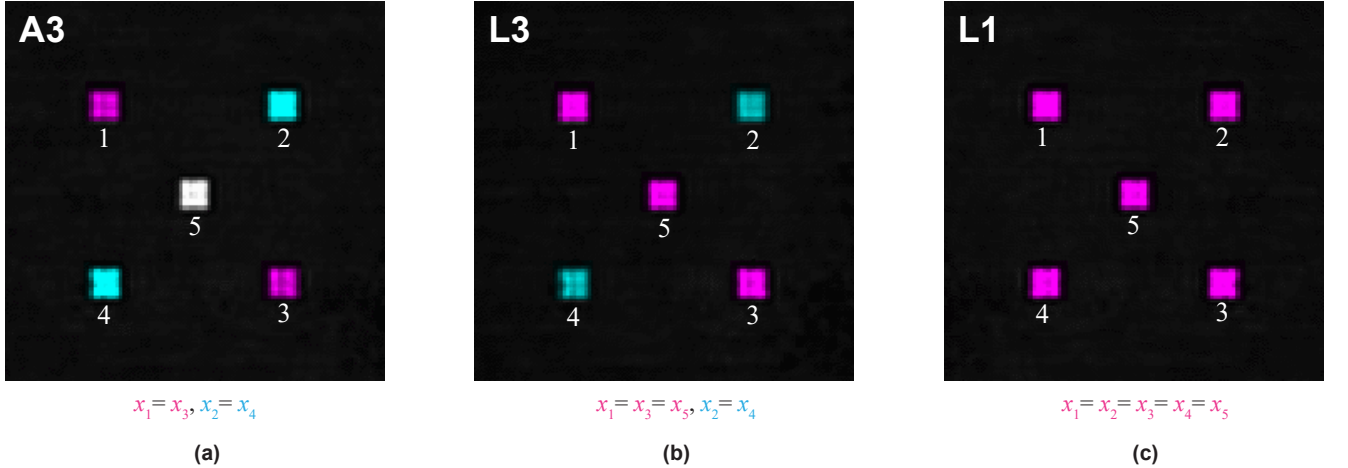


FIG. 6. Snapshots of the experimental dynamics for the cases of a) three clusters (A3), b) two clusters (L3) and c) one cluster (L1).

## V. CONCLUSION

We have studied the emergence of cluster synchronization in networks of coupled oscillators. First, we show how to obtain all of the possible dynamically valid CS patterns for an arbitrary network topology. The methods we illustrated above are general and will apply to nodes with any type of dynamics (ODEs, maps, etc.). This method also extends to directed networks, weighted networks, and labeled nodes (to represent different dynamics on each node), all of which are handled by the software package [34]. An important point is that these techniques work for any subgroups of the cluster groups which includes the trivial subgroups [24] (only the identity in each). Hence, we can approach merging

as a bottom up process and analyze any arbitrary merging of nodes into a cluster to determine if the dynamics allows a synchronized state and if it is stable in some parameter range. This means all possible clusters can be analyzed using our approach. We note that combining clusters as a “top down” approach would provide clusters that most likely are not from symmetries whereas bottom up would be a process that would include clusters easily obtained from symmetries, although it would be useful for cases where one has particular clusters in mind to analyze.

Our main result is a technique to evaluate stability of all the dynamically valid CS patterns for both networks for which the connectivity is given by an adjacency matrix and by a Laplacian matrix. We predict that the range of stability typically becomes smaller for CS patterns that are characterized by higher symmetry, which is confirmed in our experimental system.

We acknowledge help with computational group theory from Prof. David Joyner of the US Naval Academy and information and computer code from Ben D. MacArthur and Rubén J. Sánchez-García both of the University of Southampton. This work was funded by the Office of Naval Research. F.S. was supported by NSF grant CMMI-1400193.

- 
- [1] C. L. DeMarco and J. Wassner, in *Control Applications*, 1995, *Proceedings of the 4th IEEE Conference on* (IEEE, 1995) pp. 611–617.
  - [2] A. E. Motter, S. A. Myers, M. Anghel, and T. Nishikawa, *Nat. Phys.* **9**, 191 (2013).
  - [3] S. Abdollahy, O. Lavrova, A. Mammoli, S. Willard, and B. Arellano, in *Innovative Smart Grid Technologies (ISGT)*, 2012 IEEE PES (IEEE, 2012) pp. 1–6.
  - [4] S. Boccaletti, J. Kurths, G. Osipov, D. Valladares, and C. Zhou, *Physics reports* **366**, 1 (2002).
  - [5] L. M. Pecora and T. L. Carroll, *Phys. Rev. Lett.* **80**, 2109 (1998).
  - [6] A. Pikovsky, M. Rosenblum, and J. Kurths, *Synchronization: a universal concept in nonlinear sciences*, Vol. 12 (Cambridge university press, 2003).
  - [7] C. Zhou and J. Kurths, *Chaos* **16**, 015104 (2006).
  - [8] A-L. Do, J. Hoefener, and T. Gross, *New Journal of Physics* **14**, 115022 (2012).
  - [9] T. Dahms, J. Lehnert, and E. Schöll, *Physical Review E* **86**, 016202 (2012).
  - [10] C. Fu, Z. Deng, L. Huang, and X. Wang, *Physical Review E* **87**, 032909 (2013).
  - [11] I. Kanter, M. Zigzag, A. Englert, F. Geissler, and W. Kinzel, *EPL* **93**, 6003 (2011).
  - [12] D. P. Rosin, D. Rontani, D. Gauthier, and E. Schöll, *Phys. Rev. Lett.* **110**, 104102 (2013).
  - [13] F. Sorrentino and E. Ott, *Physical Review E* **76**, 056114 (2007).
  - [14] C. R. Williams, T. E. Murphy, R. Roy, F. Sorrentino, T. Dahms, and E. Schöll, *Phys. Rev. Lett.* **110**, 064104 (2013).
  - [15] O. D’Huys, R. Vicente, T. Erneux, J. Danckaert, and I. Fischer, *Chaos* **18**, 037116 (2008).
  - [16] G. Orosz, *SIAM Journal on Applied Dynamical Systems* **13**, 1353 (2014).
  - [17] I. Belykh, E. de Lange, and M. Hasler, *Physical review letters* **94**, 188101 (2005).
  - [18] A. B. Cohen, B. Ravoori, F. Sorrentino, T. E. Murphy, E. Ott, and R. Roy, *Chaos: An Interdisciplinary Journal of Nonlinear Science* **20**, 043142 (2010).
  - [19] V. Belykh, G. Osipov, V. Petrov, J. Suykens, and J. Vandewalle, *Chaos* **18**, 037106 (2008).
  - [20] G. Russo and J.-J. E. Slotine, *Physical Review E* **84**, 041929 (2011).
  - [21] M. Golubitsky and I. Stewart, *The Symmetry Perspective: From Equilibrium to Chaos in Phase Space and Physical Space*, Vol. II (Birkhäuser-Verlag, Basel, 2002).
  - [22] M. Golubitsky, I. Stewart, and D. Schaeffer, *Singularities and groups in bifurcation theory*, Vol. II (Springer-Verlag, New York, NY, 1985).
  - [23] V. Nicosia, M. Valencia, M. Chavez, A. Díaz-Guilera, and V. Latora, *Phys. Rev. Lett.* **110**, 174102 (2013).
  - [24] L. M. Pecora, F. Sorrentino, A. M. Hagerstrom, T. E. Murphy, and R. Roy, *Nature Communications* **5** (2014).
  - [25] B. Ravoori, A. B. Cohen, A. V. Setty, F. Sorrentino, T. E. Murphy, E. Ott, and R. Roy, *Physical Review E* **80**, 056205 (2009).
  - [26] M. Golubitsky, I. Stewart, and A. Török, *SIAM Journal on Applied Dynamical Systems* **4**, 78 (2005).
  - [27] V. N. Belykh, I. V. Belykh, and M. Hasler, *Physical Review E* **62**, 6332 (2000).
  - [28] V. N. Belykh, I. V. Belykh, and E. Mosekilde, *Physical Review E* **63**, 036216 (2001).
  - [29] L. Pecora and T. Carroll, *Phys. Rev. Lett.* **80**, 2109 (1998).
  - [30] H. Kamei and P. J. A. Cock, *SIAM J. Applied Dynamical Systems* **12**, 352 (2013).
  - [31] B. MacArthur and R. Sanchez-Garcia, *Physical Review E* **80**, 026117 (2009).
  - [32] B. MacArthur, R. Sanchez-Garcia, and J. Anderson, *Discrete Appl. Math.* **156**, 3525 (2008).
  - [33] T. G. Group, *GAP: Groups, Algorithms, and Programming*, Version 4.4 (<http://www.gap-system.org>, 2005).
  - [34] W. Stein, *SAGE: Software for Algebra and Geometry Experimentation* (<http://www.sagemath.org/sage/>, <http://sage.scipy.org/>, 2013).
  - [35] W. Poel, A. Zakharova, and E. Schöll, *Physical Review E* **91**, 022915 (2015).
  - [36] S. C. Manrubia and A. S. Mikhailov, *Physical review E* **60**, 1579 (1999).
  - [37] I. Belykh and M. Hasler, *Chaos: An Interdisciplinary Journal of Nonlinear Science* **21**, 016106 (2011).

- [38] A. M. Hagerstrom, T. E. Murphy, R. Roy, P. Hövel, I. Omelchenko, and E. Schöll, *Nature Physics* **8**, 658 (2012).